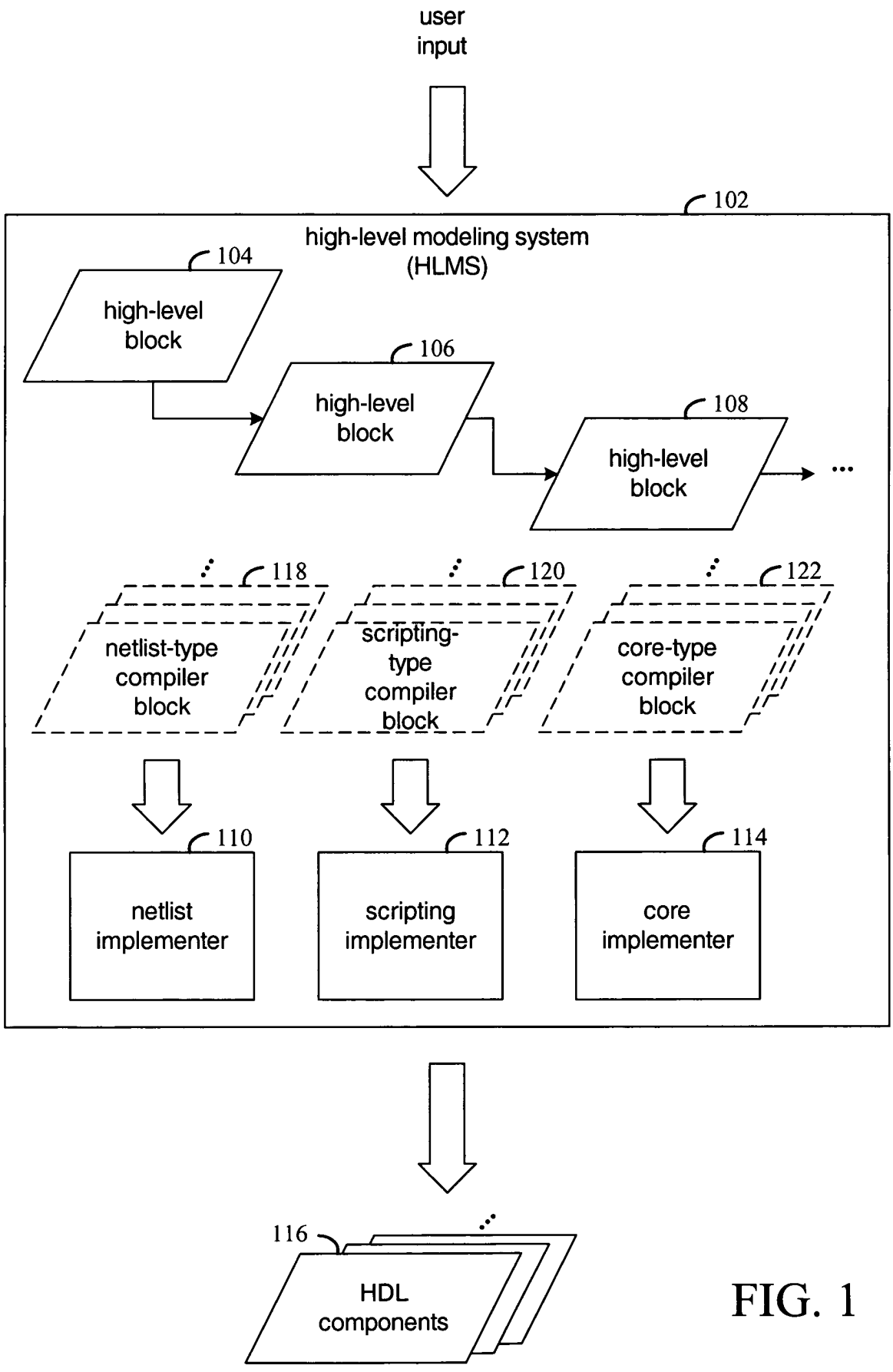
user input
102
high-level modeling system (HLMS)
104
high-level block
106
high-level block
108
high-level block
...
118
netlist-type compiler block
120
scripting-type compiler block
122
core-type compiler block
110
netlist implementer
112
scripting implementer
114
core implementer
116
HDL components

FIG. 1

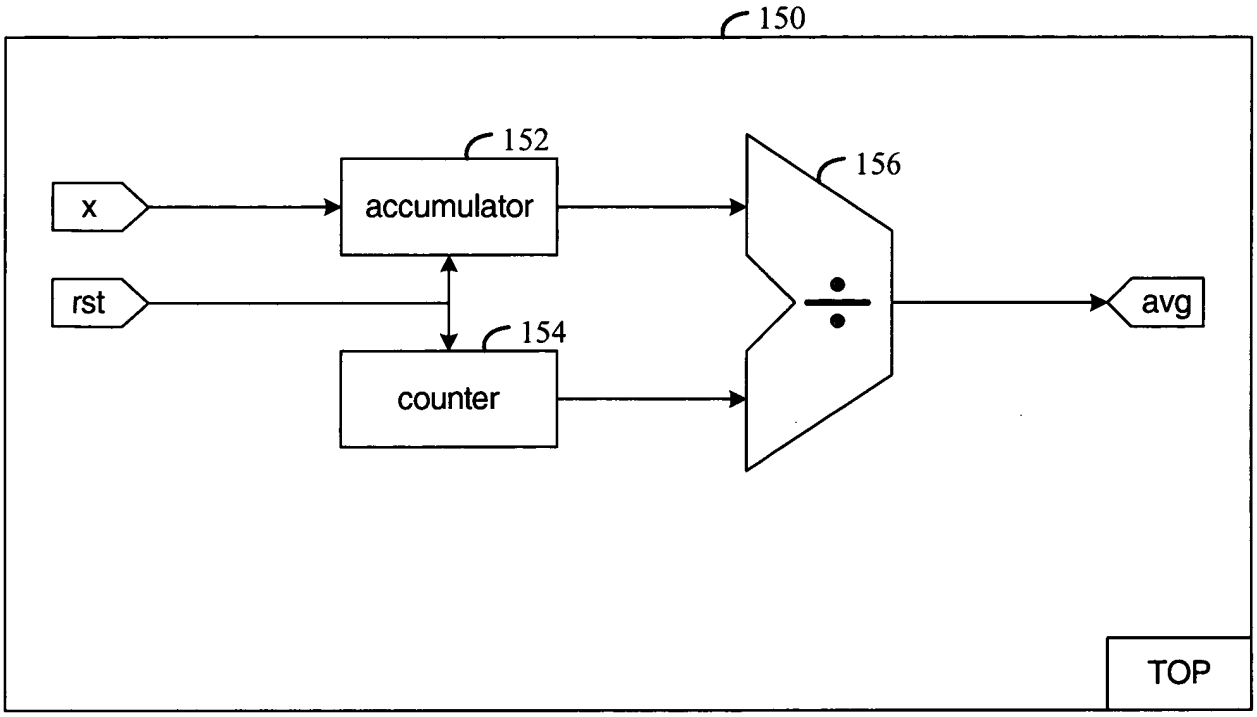
150
152
accumulator
156
x
rst
154
counter
avg
TOP

FIG. 2

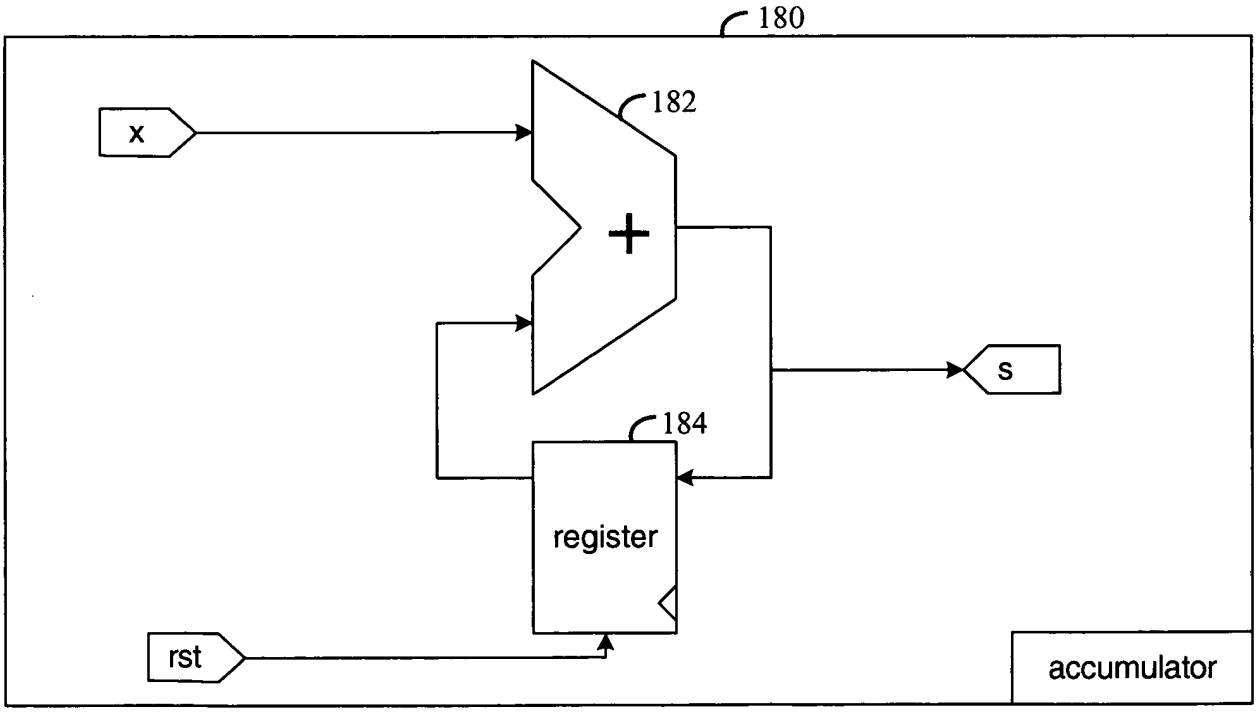
180
182
x
+
s
184
register
rst
accumulator

FIG. 3

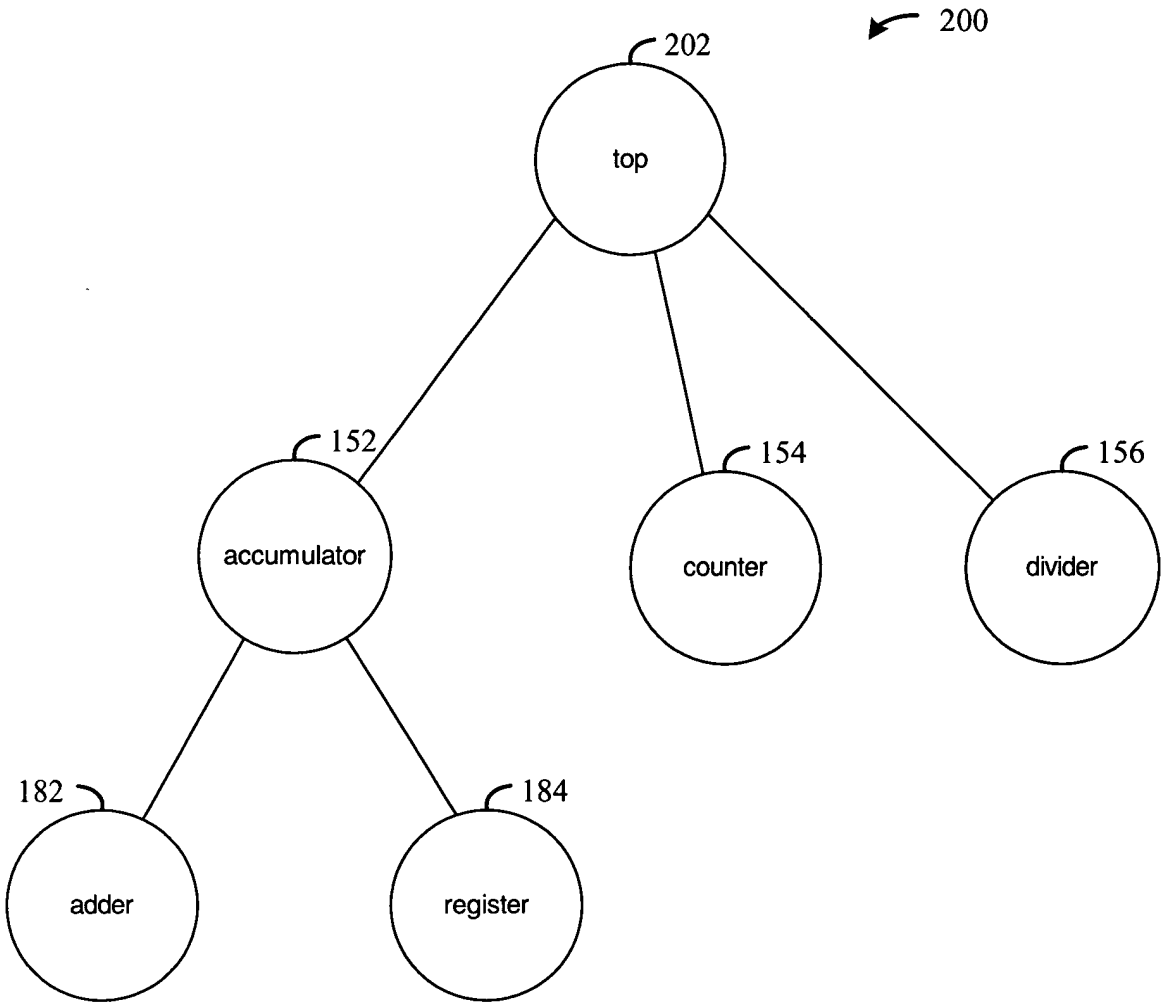
200
202
top
152
accumulator
154
counter
156
divider
182
adder
184
register

FIG. 4

250

Design-to-Hardware-Description Translation	
252	Perform necessary design-wide work on high-level blocks
254	Prepare implementers
256	b_h := top block of the HLMS design
258	Compile (b_h)
260	Complete all pending work in the implementers

FIG. 5

280

Compile (b_h)	
282	Allocate memory for a compiler block
284	Translate b_h into appropriate type of compiler block, b_k
286	If b_h has sub-blocks then
288	For each sub-block, b_i
290	Compile (b_i)
292	Implement b_k using the appropriate implementer
294	Save a lightweight description of the HDL implementation of b_k in association with the parent block of b_k
296	Free the memory of the compiler block b_k
298	Return

FIG. 6

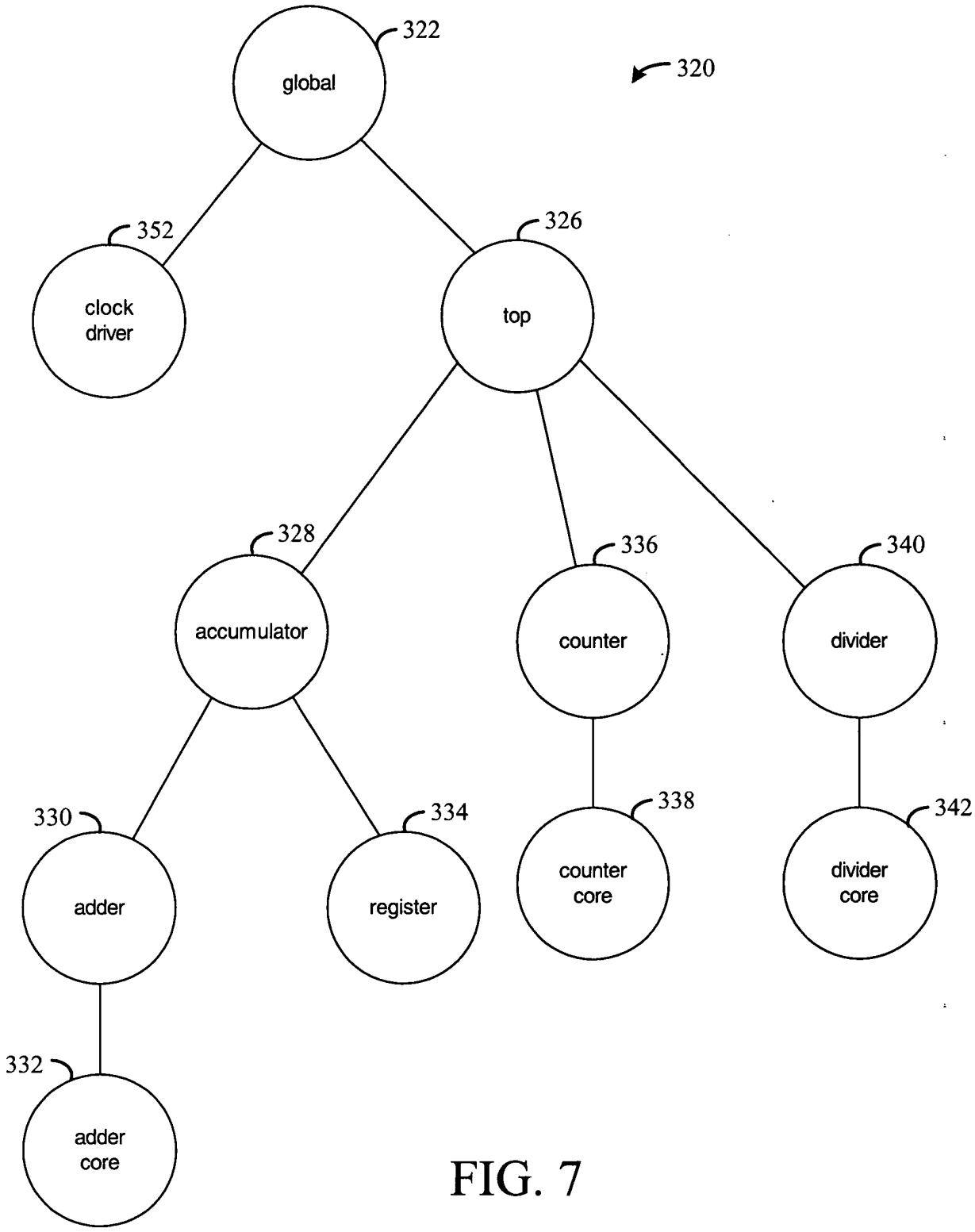
global
322
320
352
clock driver
326
top
328
accumulator
336
counter
340
divider
330
adder
334
register
338
counter core
342
divider core
332
adder core

FIG. 7

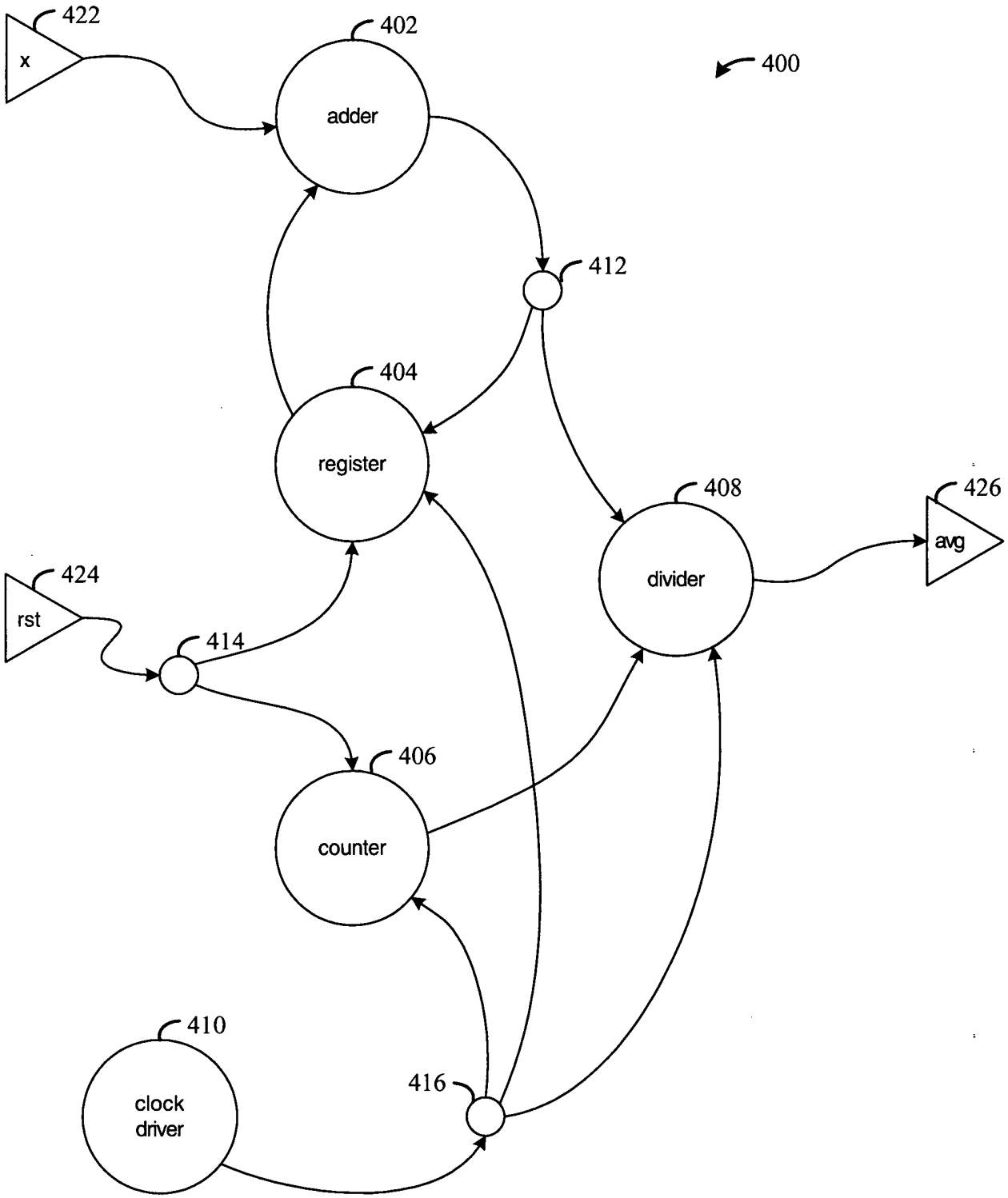
422
x
402
adder
400
412
404
register
408
divider
426
avg
424
rst
414
406
counter
410
clock
driver
416

FIG. 8